VOL. 11, NO. 5, MARCH 2016 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences


www.arpnjournals.com

TWO SECURE NON-SYMMETRIC ROLE KEY-AGREEMENT PROTOCOLS

S. Ghoreishi, I. F. Isnin, S. A. Razak and H. Chizari
Faculty of Computing, Universiti Teknologi Malaysia, Johor, Malaysia
E-Mail: kmohsen.gh100@gmail.com

ABSTRACT

Recently, some two-party Authenticated Key Agreement protocols over elliptic curve based algebraic groups, in the context of Identity-Based cryptography have been proposed. The main contribution of this category of protocols is to reduce the complexity of performing algebraic operations through eliminating the need to using Bilinear Pairings. In this paper, we proposed two novel Identity-Based Authenticated Key Agreement protocols over non-symmetric role participants without using Bilinear Pairings. The results show that our proposed schemes beside of supporting security requirements of Key Agreement protocols, require a subset of operations with low complexity in compare with related protocols in this scientific area.

Keywords: key agreement, identity-based, elliptic curves, computational complexity.

INTRODUCTION

The Key Agreement is a cryptographic primitive in which communicating participants interact with each other in an open channel to establish a secure session key. In a two-party Key Agreement protocol mentioned session key will be shared between two participants. In the context of Public Key Cryptography, Key Agreement protocols can be appeared in the form of Certificate-Based, Identity-Based or Certificateless cryptosystems. However, this paper emphasizes on Identity-Based Key Agreement protocols. The main advantage of Identity-Based cryptosystems in compare with Certificate-Based ones is solving the drawbacks related to complex management of Certification Authorities (CA) and Public Key Infrastructures (PKI). Adi Shamir in 1984 was the pioneer of the idea of Identity-Based cryptography [1].

The main contribution of the mentioned work was assuming a meaningful public key, called identity, for all entities. As a result, existing entities do not require to validate authenticity of public keys. This functionality in turn led to eliminating the need to certificates. Although this interesting solution seems to be useful in a large variety of applications, the lack of a practical Identity-Based cryptosystem caused that this scientific area remained an open problem for many years. However, Boneh and Franklin could propose a fully functional and practical Encryption scheme in the context of Identity-Based cryptography in 2001 [2].

The core function in such an applicable scheme was the use of a cryptographic function named Bilinear Pairing. The functionality of Bilinear Pairing is mapping an input which consists of two elements of elliptic curve based algebraic groups, to an element of a multiplicative group over finite fields [3]. Followed by proposing two significant pairing-based schemes, Joux's tree-party Key Agreement protocol [4] and Identity-Based Encryption by Boneh and Franklin [2], a large variety of Identity-Based schemes based on Bilinear Pairings had been proposed including Identity-based Key Agreement Protocols [5-8]. Although these protocols could support many security requirements of Key Agreement protocols [9], high cost of pairing operation made them expensive from computational complexity perspective [10-12]. To overcome this drawback, we have been proposed a subset of Identity-Based and Certificateless Pairing-Free Key Agreement protocols by the use of scalar multiplication over elliptic curve based algebraic groups instead of expensive pairing operations [13-17]. In order to propose efficient Key Agreement protocols in this category, we have presented two novel Identity-Based Pairing-Free protocols with non-symmetric role participants

This paper is organized as follows. In the following section the essential technical backgrounds are provided. In the third section, existing related works are reviewed comprehensively. Our proposed protocols are presented in the fourth section. Afterward, we analyzed our proposed Key Agreement protocols from the security viewpoint. In the section 6, the performance of our works is compared with related works. Finally, the conclusion of this paper is provided in the last section.

Technical backgrounds

This section presents the required backgrounds related to the scope of this paper. In this way, the following sub-section introduces the idea of Elliptic Curve Cryptography (ECC) [18].

Elliptic curve cryptography

Elliptic Curves are one of the fundamental scientific topics in many cryptographic literatures. Since the use of cryptographic operations over Elliptic Curve based algebraic groups is the basis of pairing-free Key Agreement protocols, this section investigates the details of this category of groups, briefly. If roughly speaking, this category of cryptographic curves are defined over a set of points and a binary operation, named addition, to generate a beneficial algebraic group. In more detail, the points of an Elliptic Curve over a finite field Fp^n are a subset of the Equation (1) by considering nonzero discriminant ($\Delta = -16[4a^3 + 27b^2]$).

$$y^2z = x^3 + axz^2 + bz^3 \quad (1)$$

Here, the coefficients a and b are the elements of considered finite field, Fp^n. Assume that (0, y_0, 0) is the identity element of mentioned algebraic group, remained elements would be the solutions of the Equation.

$$y^2 = x^3 + ax + b \quad (2)$$

In order to explain the functionality of the addition operation over the mentioned curve above, two samples of points are considered. Without loss of generality, assume that two points (x_1 , y_1) and (x_2, y_2) are considered and (x_1, y_2) + (x_2, y_2) = (x_3, y_3). In this condition, the result is (x_3, y_3) = (λ^2-x_1-x_2, λ (x_1-x_3) –y_1). The value of λ can be calculated as Equation (3).

$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1}, (x_1, y_1) \neq (x_2, y_2) \\ \frac{3x_1^2 + a}{2y_1}, (x_1, y_1) = (x_2, y_2) \end{cases} \quad (3)$$

A REVIEW OF IDENTITY-BASED AUTHENTICATED KEY AGREEMENT PROTOCOLS

A subset of existing Pairing-Free Key Agreement protocols in the context of Identity-Based cryptography are reviewed in this section. The details of standardized Setup and Extraction phases of existing protocols are as followed.

Setup: In this phase, PKG takes the required security parameter, and returns the confidential Master-Key $s \in \mathbb{Z}_q^*$ and system parameters Params $< q, \mathbb{F}_q, E/\mathbb{F}_q, G, P, P_{Pub}, H_1, kdf >$ which is publicly known to all entities. In this tuple, q is a large prime number, $\mathbb{F}_q$ is a finite field over q, and $E/\mathbb{F}_q$ is an elliptic curve over $\mathbb{F}_q$. Moreover, P refers to the generator of the group G (a subgroup of $E/\mathbb{F}_q$). Finally, $H_1: \{0,1\}^* \times G \rightarrow \mathbb{Z}_q^*$ is a collision free hash functions and kdf is a key derivation function which maps considered input to the determined number of bits as the session-key.

Extraction: In this phase, each entity such as i who possesses ID_i identifier refers to PKG to take corresponding Private Key. Here, PKG randomly chooses $r_i \in_r \mathbb{Z}_q^*$, then computes $R_i = r_iP$ and $h_i = H_1(ID_i, R_i)$. Afterward, PKG transfers $< R_i, s_i >$ to the entity. After taking mentioned transferred message, the entity computes the value $h_i = H_1(ID_i, R_i)$, then verifies authenticity of received private key, s_i, by checking the equality of the Equation (4).

$$s_iP = R_i + h_iP_{Pub} \quad (4)$$

In continue, other phases of the considered Identity-Based Key Agreement protocols are explained separately. The final session-key would be the output of kdf function of agreed shared values and a subset of publicly known or transferred items.

The Figure-1 depicts the EXCHANGE and COMPUTATION phases of the proposed Pairing-Free Identity-Based Key Agreement protocol by Cao et al. [19]. After computing agreed values, the session key will be calculated as (5)

$$k_s = kdf(K_{AB}^1, K_{AB}^2) \quad (5)$$

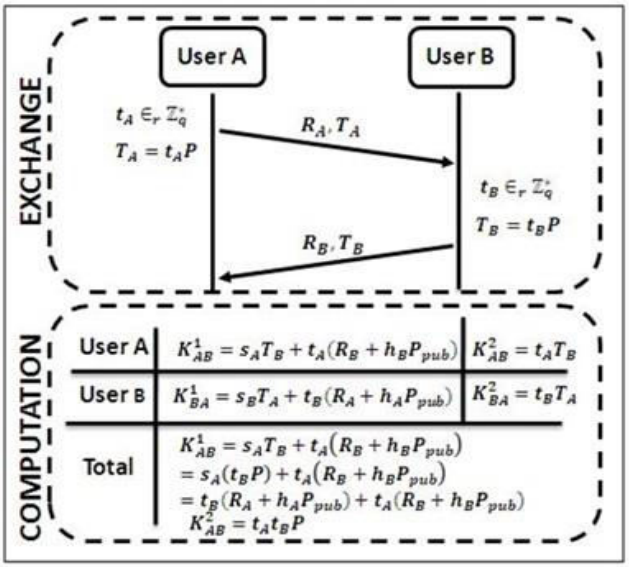

Figure-1. Exchange and Computation phases of Key Agreement protocol proposed by Cao *et al.* [19].

In addition, Islam et al. proposed another protocol in this category, which is demonstrated in the Figure-2 [20]. Here, the session key would be computed such as equation (6).

$$k_s = kdf(K_{AB}) \quad (6)$$

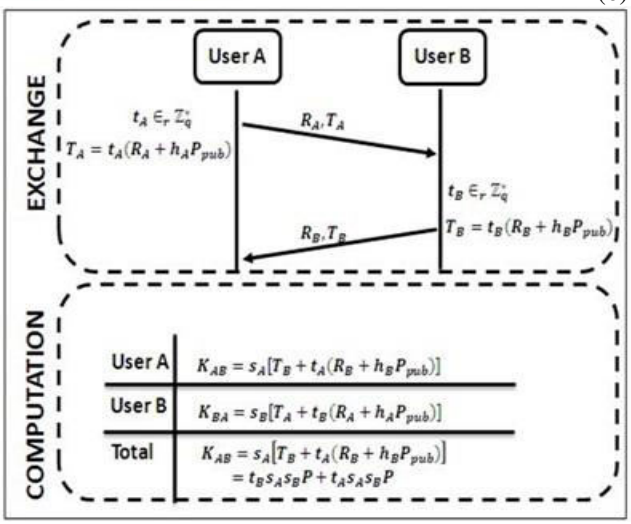

Figure-2. Exchange and Computation phases of Key Agreement protocol proposed by Islam *et al.* [20].

Finally, the Figure-3 depicts the Exchange and Computation phases of proposed Pairing-Free Identity-

Based Key Agreement protocol by Farrash *et al.* [21]. After computing agreed values, the session key will be calculated as (7).

$$k_s = kdf(K_{AB}^1, K_{AB}^2) \quad (7)$$

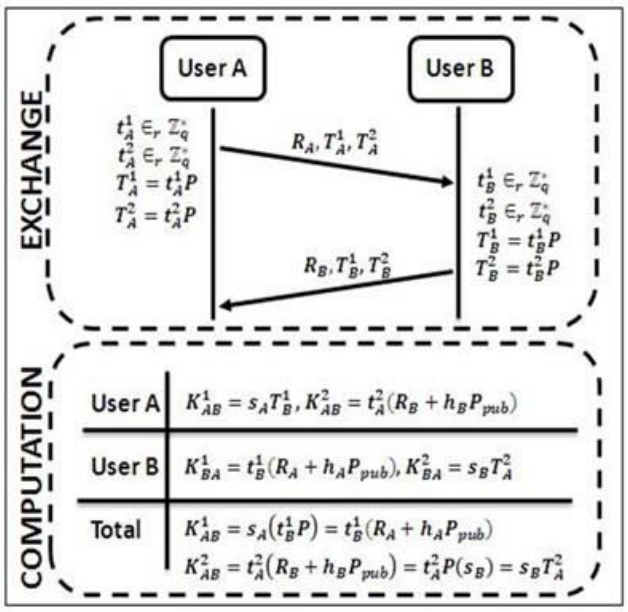

Figure-3. Exchange and Computation phases of key agreement protocol proposed by Farash *et al.* [21].

OUR PROPOSED KEY AGREEMENT PROTOCOLS

This section describes our two proposed protocols in detail. These protocols are non-symmetric role, more precisely performing EXCHANGE phase by the first participant and second one are not the same [14].The details of SETUP and EXTRACTION phases of our proposed protocols are the same as what have been explained about these two phases in the previous section. Other phases of our proposed protocols are explained separately as followed.

First proposed protocol

Assume that two entities, A and B, are going to agree on a session-key. It is necessary to point out that the entity B (second participant), randomly choose $p_B \in_r \mathbb{Z}_q^*$, then computes $P_B = p_B P$, $q_B = p_B + h'_B s_B (\text{mod q})$, and $Q_B = q_B P$. Here, $h'_B = H_1(ID_B, P_B)$.

Exchange: To explain the EXCHANGE phase, mentioned entities do the following:

(1) A chooses a random $t_A \in_r \mathbb{Z}_q^*$, computes the key token $T_A = (t_A s_A)P = t_A((r_A + h_A s(\text{mod q}))\text{P})$ and sends T_A, R_A to the B entity.

(2) B chooses a random $t_B \in_r \mathbb{Z}_q^*$, computes the key token $T_B = (t_B q_B)P = t_B((p_B + h'_B s_B(\text{mod q}))\text{P})$ and sends T_B, R_B, P_B to the A entity.

Computation: In this phase, the entities A and B are able to compute the shared secret as follows:

A computes $K_{AB} = [t_A s_A]T_B$

B computes $K_{BA} = [t_B(p_B + h'_B s_B(\text{mod q}))]T_A$

Following equation proves that the two computed values for this shared secrets would be the same.

$$\begin{aligned} K_{AB} &= [t_A(r_A + h_A s(\text{mod q}))]T_B \\ &= (t_A s_A)[t_B((p_B + h'_B s_B(\text{mod q}))\text{P})] \\ &= (t_A s_A)(t_B q_B)P \\ &= [t_B(p_B + h'_B s_B(\text{mod q}))]T_A = K_{BA} \end{aligned}$$

Second proposed protocol

Assume that two entities, A and B, are going to agree on a session key. It is necessary to point out that the entity A, randomly chooses $p_A \in_r \mathbb{Z}_q^*$, then computes $P_A = p_A P$, $q_A = p_A + h'_A s_A (\text{mod q})$, and $Q_A = q_A P$. Here, $h'_A = H_1(ID_A, P_A)$.

Exchange: To explain the EXCHANGE phase, mentioned entities do the following:

(1) A chooses a random $t_A \in_r \mathbb{Z}_q^*$, computes the key token $T_A = (t_A q_A)P = t_A((p_A + h'_A s_A(\text{mod q}))\text{P})$ and sends T_A, R_A, P_A to the B entity.

(2) B chooses a random $t_B \in_r \mathbb{Z}_q^*$, computes the key token $T_B = (t_B s_B)P = t_B((r_B + h_B s(\text{mod q}))\text{P}$ and sends T_B, R_B to the A entity.

Computation: In this phase, the entities A and B are able to compute the shared secret as follows:

A computes $K_{AB} = [t_A(p_A + h'_A s_A(\text{mod q}))]T_B$

B computes $K_{BA} = [t_B s_B]T_A$

Following equation proves that the two computed values for this shared secrets would be the same.

$$\begin{aligned} K_{AB} &= [t_A(p_A + h'_A s_A(\text{mod q}))]T_B \\ &= (t_A q_A)[t_B((r_B + h_B s(\text{mod q}))\text{P})] \\ &= (t_A q_A)(t_B s_B)P \\ &= [t_B(r_B + h_B s(\text{mod q}))]T_A = K_{BA} \end{aligned}$$

SECURITY EVALUATION

In this section we investigate the security requirements followed by what mentioned in [22, 23]. Without loss of generality, we investigated our first protocol. The results will be the same for second one.

Before investigating the security requirements below, it is worth to note that considered adversary is able to reach the values $S_i = s_i P$, P_i , Q_i and $T_i$related to an entity who possesses ID_iidentifier through eavesdropping an open channel or computing some values from some other possessing ones.

Known-key security: Through the renewal process of computing t_A and t_B, the value of session-key in new session would be unique and independent in compare with the value of session-key in last session. Therefore, any subset of possessing session-keys does not help the adversary to compute the next session-key.

Forward secrecy: By assuming that adversary compromises the values s_A and s_B, it is not possible to reach the value of session-key. Obviously, computing the value of the shared session-key, $(t_A t_B q_A s_B)P$, requires to reach the values t_A or t_B. However, the discrete logarithm problem makes the mentioned adversary unable to compute the final value of session-key.

Perfect forward secrecy: Here, it is assumed that the adversary knows the value of Master-key, s, beside of the values s_A and s_B. The same reasons mentioned in "forward secrecy" property are sufficient to prove that the proposed protocol(s) support(s) "perfect forward secrecy" property.

Key-compromise impersonation: An adversary who possesses the value $s_A$is unable to impersonate the entity B to A. the reason is that possessing the values s_A and $T_A = (t_A s_A)P$ leads to reach value $t_A P$. By assuming that adversary is able to compute the value $T_B = (t_B q_B)P$, it is impossible to compute the value of final session-key, $(t_A t_B q_A s_B)P$. The reason is that computing the value $(t_A q_B)P$ which can be driven from session-key (because adversary possesses the values s_A and t_B), while adversary possesses the values $t_A P$ and $q_B P$ (and not t_A and q_B) requires solving one of the mathematical hard problems named Computational Diffie Hellman (CDH) which is impractical in polynomial time.

Unknown key-share resilience: Here, it is assume that adversary possesses the publics, while tries to impersonate entity B to A. Mentioned reasons for supporting "key-compromise impersonation" in which adversary was able to reach more values (includings_A) is sufficient to support "unknown key-share resilience".

Key control: This property emphasizes on making communicating parties in predetermination of the session-key. Since, the value of session-key in any session, $(t_A t_B q_A s_B)P$, depends on the values t_A and t_B, a cheating participant is unable to predetermine the value of session-key in the next sessions.

Unknown session-specific temporary information: Here, it is assumed that the vales t_A and t_B are leaked and the adversary knows them. Hence, computing the value $(s_A q_B)P$ is enough for the adversary who just possesses the values $s_A P$ and $q_B P$ (and not s_A and q_B). Computation of mentioned values above requires solving one of the mathematical hard problems named Computational Diffie Hellman (CDH) which is impractical in polynomial time.

PERFORMANCE COMPARISONS

This section shows the excellence of our proposed protocols in compare with other existing related works. Since improving the efficiency is one of the challenging issues for key agreement protocols [24] this section shows that our proposed scheme is considerably more efficient than other proposed ones in this scientific are. The focus of this comparison is on the total computational cost during the increase in the number of established sessions. The main reason for proposing such a comparison is that in the considered works, there exist some computations which must be performed once.

Hence, when the number of established sessions is grown, the overall computational cost of the mentioned works will be different. Before starting the comparison based on this method, it is important to learn about the computational costs of operations. Table 1 represents computational costs of various operations [25]. In this table, the complexity of executing modular multiplication is considered as a meter of estimating other operations' complexities.

Table-1. Computational costs of operations [25].

Notation	Definition and Conversion
T_{MM}	Time complexity for executing the modular multiplication
T_{SM}	Time complexity for executing the elliptic curve scalar point multiplication $1T_{SM} \approx 29T_{MM}$
T_{PA}	Time complexity for executing the elliptic curve point addition , $1T_{PA} \approx 0.12T_{MM}$
T_{IN}	Time complexity for executing the modular inversion operation, $1T_{IN} \approx 11.6T_{MM}$

Based on what explained above and given information in Table-1, Table-2 shows the computational complexity of utilized operations as a function of the number of established sessions in the considered works. Based on Table-2, Figure-4 illustrates the difference of overall computational costs among considered works during the growth of number of established sessions. It is apparent from the Figure-4 that our work is effectively less complex than the other ones in such condition.

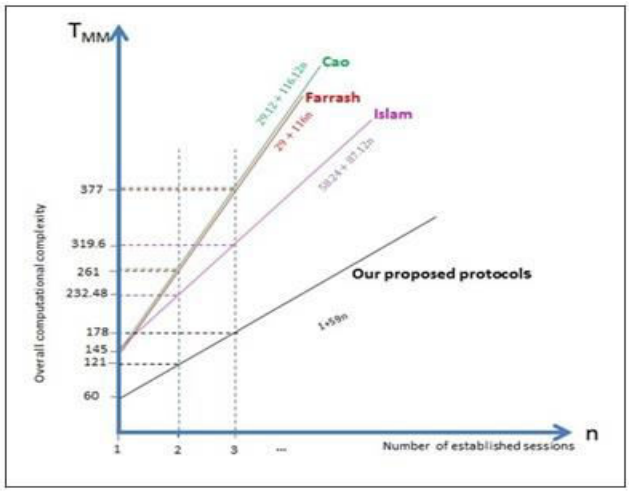

Figure-4. Computational complexity growth of considered protocols.

Table-2. Performance comparison during the growth of number of the established sessions.

Protocols	Cao et al. [19]	Islam et al. [20]	Farrash et al. [21]	Our First Protocol	Our Second Protocol
Pre Computed Scalar Multiplication	$h_B P_{pub}$	$h_A P_{pub}$, $h_B P_{pub}$	$h_B P_{pub}$	–	–
Pre Computed Modular Multiplication	–	–	–	$h_A s \pmod q$	$h'_A s_A \pmod q$
Pre Computed Point Addition	$R_B + h_B P_{pub}$	$R_A + h_A P_{pub}$, $R_B + h_B P_{pub}$	$R_B + h_B P_{pub}$	–	–
Performance Consideration	$5T_{SM} + 2T_{PA} + (n-1)[4T_{SM} + T_{PA}] = T_{SM} + T_{PA} + n[4T_{SM} + T_{PA}] = 29 + 0.12 + n[4(29) + 0.12] = 29.12 + 116.12n$	$5T_{SM} + 3T_{PA} + (n-1)[3T_{SM} + T_{PA}] = 2T_{SM} + 2T_{PA} + n[3T_{SM} + T_{PA}] = 2(29) + 2(0.12) + n[3(29) + 0.12] = 58.24 + 87.12n$	$5T_{SM} + T_{PA} + (n-1)[4T_{SM}] = T_{SM} + n[4T_{SM}] = 29 + 116n$	$2T_{SM} + 2T_{MM} + (n-1)[2T_{SM} + T_{MM}] = T_{MM} + n[2T_{SM} + T_{MM}] = 1 + n[2(29) + 1] = 1 + 59n$	$2T_{SM} + 2T_{MM} + (n-1)[2T_{SM} + T_{MM}] = T_{MM} + n[2T_{SM} + T_{MM}] = 1 + n[2(29) + 1] = 1 + 59n$

CONCLUSIONS

Nowadays, pairing-free cryptography became an active scientific topic especially in the area of Authenticated Key Agreement protocols. The main reason is that Bilinear Pairings impose high complexity of computations which leads to lower performance in compare with pairing-free ones. In this paper, we could propose two novel Identity-based two-party Key Agreement protocols over Elliptic Curves that have better performance in compare with existing related works from the viewpoint of overall complexity of computing operations.

ACKNOWLEDGEMENTS

The Authors would like to thank Universiti Teknologi Malaysia and Ministry of Higher Education, Malaysia for sponsoring this research under vote number Q.J130000.2513.08H30.

REFERENCES

[1] Shamir. 1984. Identity-Based Cryptosystems and Signature Schemes. In Advances In Cryptology—Crypto 1984, Lecture Notes In Comput. Sci. 196, Springer-Verlag, Berlin.

[2] D. Boneh, M. Franklin. 2001. Identity Based Encryption From The Weil Pairing. Advances in Cryptology-Crypto.

[3] S. Galbraith, K. Paterson, N. Smart. 2008. Pairings for Cryptographers. Discrete Applied Mathematics. 156(16): 3113–3121.

[4] A. Joux. 2000. A One Round Protocol for Tripartite Diffie-Hellman. Proceedings of ANTS 4, LNCS1838. 385-394.

[5] N.P. Smart. 2002. An identity based authenticated key agreement protocol based on the Weil pairing. Electro. Lett. 38: 630–632.

[6] L. Chen, C. Kudla. 2003. Identity based authenticated key agreement from pairings. In: IEEE Computer Security Foundations Workshop. 219–233.

[7] Q. Yuan, S.A. Li. 2005. A new efficient ID-based authenticated key agreement protocol. Cryptology ePrint Archive, Report.

[8] Y. Wang. 2013. Efficient Identity-Based and Authenticated Key Agreement Protocols. Transactions on Computational Science Xvii.

[9] L. Chen, Z. Cheng, N.P. Smart. 2007. Identity-based key agreement protocols from pairings. International Journal Information Security. 6: 213–241.

[10] D. He, C .Chen, S .Chan, J. Bu. 2012. Secure and efficient handover authentication based on bilinear pairing functions. IEEE Transactions on Wireless Communications. 11(1): 48–53.

[11] D. Aranha, K. Karabina, P. Longa, C. Gebotys, J. López. 2011. Faster explicit formulas for computing pairings over ordinary curves. Lecture Notes in Computer Science. 6632: 48–68.

[12] S. M. Ghoreishi, I. F. Isnin. 2013. Secure Lightweight Pairing-Based Key-Agreement Cryptosystems: Issues and Challenges. IACSIT International Journal of Engineering and Technology. 5 (2).

[13] S. M. Ghoreishi, S. Abd Razak, I. F. Isnin, H. Chizari. 2014. New Secure Identity-Based and Certificateless Authenticated Key Agreement protocols without Pairings. In: Proceedings of 2014 International

Symposium on Biometrics and Security Technologies (ISBAST), Kuala Lumpur, MALAYSIA. pp. 188-192.

[14] S. M. Ghoreishi, I. F. Isnin, S. Abd Razak, H. Chizari. 2014. A novel secure two-party Identity-Based Authenticated Key Agreement protocol without Bilinear Pairings. In: Proceedings of 4th World Congress on Information and Communication Technologies (WICT), Malacca, MALAYSIA, pp. 251-258.

[15] S. M. Ghoreishi, I. F. Isnin, S. Abd Razak, H. Chizari. An Efficient Pairing-free Certificateless Authenticated Two-party Key Agreement protocol over Elliptic Curves. In: Proceedings of 4th World Congress on Information and Communication Technologies (WICT), Malacca, MALAYSIA. 259-266.

[16] S. M. Ghoreishi, I. F. Isnin, S. Abd Razak, H. Chizari. 2015. Secure and Authenticated Key Agreement Protocol with Minimal Complexity of Operations in the Context of Identity-Based Cryptosystems. In: Proceedings of 2015 International Conference on Computer, Communication, and Control Technology (I4CT), Kuching, Malaysia.

[17] S. M. Ghoreishi, I. F. Isnin, S. Abd Razak, H. Chizari. 2015. A performance improved certificateless key agreement scheme over elliptic curve based algebraic groups. In: Proceedings of International Conference on Intelligent and Interactive Computing (IIC 2015), Melaka, Malaysia.

[18] D. Hankerson, A. J. Menezes, and S. Vanstone. 2004. Guide to Elliptic Curve Cryptography. Springer-Verlag.

[19] X. Cao, W. Kou, Y. Yu, R. Sun. 2008. Identity-based authentication key agreement protocols without bilinear pairings, IEICE Transaction on Fundamentals, E91−A (12):3833−3836.

[20] S. K Hafizul Islam, G. P. Biswas. 2012. An improved pairing-free identity-based authenticated key agreement protocol based on ECC. Procedia Engineering, 30: 499-507, ISSN 1877-7058.

[21] M. S. Farrash, M. A. Attari. 2014. A pairing-free ID-based key agreement protocol with different PKGs. Int. J. Network Security. 16(2), 143–148.

[22] Z. Cheng, M. Nistazakis, R. Comley, L. Vasiu. 2005. on the in distinguishability-based security model of key agreement protocols-simple cases. Cryptology ePrint Archieve, Report 2005/129.

[23] S. Blake-Wilson, D. Johnson, A. Menezes. 1997. Key agreement protocols and their security analysis. Proc. of the 6th IMA International Conference on Cryptography and Coding, LNCS, Springer-Verlag. 1335:30−45.

[24] S. M. Ghoreishi, I. F Isnin. 2013. Secure lightweight pairing-based key-agreement cryptosystems: Issues and Challenges. IACSIT International Journal of Engineering and Technology. 5(2).

[25] S. H. Islam, G. P. Biswas. 2012. A pairing-free identity-based authenticated group key agreement protocol for imbalanced mobile networks. Ann. Telecommun. 67 (11–12): 547–558.